NWG/RFC# 592 RWW 20-NOV-73 16:42 20391
Some Thoughts on System Design to Facilitate Resource Sharing

TYPED ONLINE: Indexed 12/5/73
FILENAME:
PROOFED:

INTRODUCTION 1

There is a growing interest in moving toward more resource sharing on the ARPANET. Some resource sharing has been taking place by having systems open TELNET connections and generating user command strings. I think that this is fine for experimental use, but is not the way we want to operate in real usage. What I believe network system builders should do is to develop mechanisms appropriately designed for computer-computer communication. 1a

SYSTEM INTERCONNECTION, AN APPROACH 2

The goal I would like to see us move toward is to view all systems on the network as offering certain service modules, any subset of which can be combined in building other systems. Each service module would have a well advertised set of primitive service capabilities that it could provide. It would have documented commands at the level of present Telnet or FTP commands for gaining access to its services. It would also have a defined network connection procedure. Then any system builder wanting to avail himself of these services could do so and integrate them into his own user interface environment. 2a

At the present time when a system is built, the system builders tend to see it as a stand alone thing or at most something to be used within a specific environment. What I would like to see fostered is the idea that any system built is not only a stand alone environment but also a network service or set of services. The builders would define not only a user interface for their environment, but also a set of primitives and primitive commands that can be accessed by other systems around the network to get that service performed. 2b

For example, we are redesigning the NLS Journal in light of our experience and that of Network Mail as a set of protocols and services. If one looks at the processes of the NLS Journal one can see a number of separate services that could be provided by different network sites or combined in varying combinations by a single site. These being: 2b1

Distribution (identification of addressees and maintainance of the required data bases being a related service), recording (numbering and storing of items), cataloging, and retrieval. 2b1a

At the moment these services are fairly tightly interconnected in the NLS Journal and what we want to do is to decouple them and define their intercommunication by protocols that would allow them to be distributed in different hosts on the network. Mechanisms

would also be defined for the several hosts performing similar services around the network to work together cooperatively. 2c

As a further example, there are also other services that NLS could probably provide such as structured file creation and manipulation; information portrayal online or in hardcopy; database querying etc. However, at the moment the system is not explicitly structured from the point of view that outside systems could come into it anywhere but at the human user interface even though internally it is quite modular. It would be straightforward for us to identify those NLS services that other system builders might possibly be interested in incorporating into their systems with their own user interface and then to do the restructuring and primitive command definition necessary. Other groups building systems on the network could perform a similar examination. 2d

CCA, on the other hand as I understand it, has taken this point of view from the beginning, namely building the Datacomputer on the assumption that it is primarily a network resource and is to be usd by other systems. BBN is also moving in this direction in the design of Distributed TENEX. 2e

There is nothing new in the above ideas; they come from generalizing past successes we have all had with network protocol development and with good software engineering practices. It will, however, take a change in the thinking of system designers, some concrete examples, and ongoing dialog to make such a design philosophy the normal network way of life. 2f

SOME FUNCTIONS READY FOR INTERCONNECTION 3

The area of dialog support may be the first area ripe to create such a synthesis with the several systems in or coming into existence, each solves part of the problem (with some overlap). The dialog support systems on the network known to me are: 3a

The NLS Journal (supports recorded and cataloged dialog and linked networks of documents and messages). 3a1

NLS Screen linking and splitting (supports close collaboration of two or more people working together in real time in NLS) 3a2

The network wide linking of terminals through BBN's RSEXEC. 3a3

Tenex Sndmsg and Readmail and other mail systems support nonrecorded dialog and further manipulation of received messages. (Some interconnection between NLS and these facilities has been established). 3a4

The communication system under design at USC-ISI to support a range of message services. 3a5

The online conferencing system being built by Jim Calvin of Case, John Iseli of Mitre and others supports online conferencing of several members and has facilities to utilize various Tenex subsystems such as TECO and NLS to support conferees. 3a6

The Hack system of CASE offers a bulletin board service. 3a7

The Forum system of IFF supports online and distributed in time conferencing and other features. 3a8

Other areas possibly ripe for synthesis are 1) file and data management, and information retrieval services; 2) editing and hardcopy portrayal with systems like Tenex RUNOFF, SU-AI's PUB and SRI-ARC's Output Processor. 3b

If the salient service features, concepts, goals of each could be defined clearly and appropriate service primitives, as per other ARPANET protocols, could be defined for each, anyone wishing to incorporate that service with a user interface appropriate to his environment or philosophy could do so. 3c

SYSTEM INTERCONNECTION ISSUES RELATED TO THE ABOVE PROPOSAL 4

There are many detailed issues related to system interconnection as proposed above. A number seem worth mentioning here. 4a

1) Types of Network Connections 4b

The number and type of network connections to be opened between classes of cooperating processes can probably be systematized. One of the important elements of the FTP and Graphics protocol efforts was to define the number and type of connections necessary for these classes of transaction. Similar classification and connection definition will be required for other types of processes. 4b1

2) Data Structure Translation 4c

The whole area of translation and transfer of data structures more complicated than sequential files needs vigorous thought and protocol development. 4c1

Systems built around sequential files are presently dominant on the ARPANET and provide a base for simple useful economical tools. I, however, do not believe that the

longer run tool sharing can depend on communication between sequential files, but requires structured files. Experience with NLS tree structured files shows that even this level of structuring may be inadequate for many uses and more sophistication may be required. A similar trend exists in work with computer graphics and generalized data management systems. Developing protocols for handling structured data bases or agreement on common structuring characteristics seems an important need. 4c1a

3) Responsiveness 4d

Factors influencing responsiveness to users in an environment of heavy geographically separated resource sharing need determination and discussion. 4d1

4) Documentation of System Interfaces 4e

It is probably resonably straightforward to define service interfaces, but they will be useless unless their activating command languages and other conventions are well documented and this documentation is kept up to date. 4e1

5) Accounting 4f

A very difficult problem once you interconnect systems at lower levels is to design an appropriate network accounting and banking system that will not cause undue delays in accessing distributed resources. 4f1

6) Error Handling 4g

We need to develop mechanisms for passing error signals around when system environments are crossing machine boundaries. 4g1

7) Standard Parameter Formats 4h

Data types such as strings, integers, floating point numbers, arrays, pointers, etc. need to have standard representations defined for passing parameters back and forth between machines. 4h1

8) HELP at the Procedure Call Level 4i

A HELP mechanism needs to be defined in the protocols to provide information that each designer can translate to his user interface. Standards for requesting HELP information and structuring HELP data bases needs agreement. 4i1

ACKNOWLEDGEMENT 5

I wish to acknowledge the useful suggestions of Charles Irby and Jim White in the thoughts above.